STPC
Video Pipeline Driver Writer's Guide

Issue 1.0

September 1999

STMicroelectronics

Technoparc du Pays de Gex - B.P. 112
165, rue Edouard Branly
01630 Saint Genis Pouilly (France)

1. INTRODUCTION

The STPC Video Pipeline is a video overlay which takes one 2D surface from the Frame Buffer area of the Unified Memory Architecture to mix it in real time with the VGA screen. Video Pipeline is usually used to display video on screen but can be use to display any type of static or dynamic image in one of the following supported formats. The Video Pipeline supports both 16 bits per pixel RGB format used for graphics and YUV422 format used for video.

The Video Pipeline can mix the input image with the graphic screen using the Colorkey method (a specific color in graphic screen is replaced by video) or the Chroma Key method (a specific range of colors in the video is replaced by the graphic screen).

The Video Pipeline can scale the input image before mixing it with the graphic screen. The current version of the Video Pipeline only supports up scaling. To preserve the input image quality, special interpolations can be used to replace the missing pixels by pixel replication.

2. SETTING UP OF SOURCE PARAMETERS

The Video Pipeline gets data from memory using 4 registers:

The Source Base Address Register

The Source Pitch Register

The Source Width Register

The Source Height Register

The Base Address corresponds to the offset in the Frame Buffer of the first pixel of the display. The Pitch corresponds to the number of Bytes to go from the first pixel of one line to the first pixel of the next line. The Width and Height Regiters correspond to the number of pixels to display.

When modifying the source or the destination size, the Scale Register has to be adjusted as described in Section 4.

3. VIDEO PIPELINE WINDOW LOCATION

Four registers describe the location of the Video Pipeline window on screen: Destination X, Y Register and Destination Width and Height Registers. After modifying Destination Width and Height Registers, the Scale Register has to be refreshed as described in Section 4.

4. SET UP SCALE REGISTER

After modifying Source or Destination size, the Scale Register has to be updated using this formula:

Horizontal scale value = (Source Width * 4096) / Destination Width

Vertical scale value = (Source Height * 4096) / Destination Height

5. SET UP THE DISPLAYING PARAMETERS

5.1. INPUT FORMAT

Video Pipeline can display 3 formats of input pixels using 16 bits per 1 pixels:

RGB555: the 5 first pixels describe the red value, 5 next pixel the green value, 5 next pixels the blue value and the last pixel is unused

RGB565: the 5 first pixels describe the red value, 6 next pixel the green value, 5 last pixels the blue value

YUV422: this format uses 2 consecutive 16 bit values to describe 2 pixels, in the following this way:

First 8 bits = Y1 value

Next 8 bits = U value

Next 8 bits = Y2 value

Next 8 bits = V value

The first pixel uses this setting: Y1UV

The second pixel uses this setting: Y2UV

This means that two consecutive pixels use the same U and V component (chrominance) and only the Y value (luminance) changes.

The input pixel format is set up in Source Pitch Register bit 12 and 13 and the color space converter has to be enabled in Color Space Converter Register in the case of YUV format.

5.2. VERTICAL FLIP OF THE IMAGE

Bit 11 of the Source Pitch Register can be put to 1 to flip image in vertical direction. In this case, the Source Base Address has to be put at the bottom of the source image instead of the top.

5.3. MIX MODE

The Mix Mode Register defines the way the Video Pipeline image is mixed with the VGA screen. In video only mode, all the surface of the Video Pipeline image is displayed and replaces the VGA screen pixels. In Color Key mode, only a specific color of the VGA screen (set in the Color Key

Register) is replaced by Video Pipeline pixels. In Chroma Key mode, only Video Pipeline pixels included in a range of RGB or YUV value are displayed and VGA screen pixels are keeped for the non matching values.

6. SCALER INTERPOLATOR

The Video Pipeline provides filters to smooth the displayed image after zooming. This filter can be enabled or disabled with bit 19 and 20 of Horizontal and Vertical Scale Register.

7. ENABLE AND DISABLE VIDEO PIPELINE

The Video Pipeline can be enabled or disabled using the bit 31 of the Status Register.

8. USE VIDEO PIPELINE IN COMBINATION WITH OTHER HARDWARE

The Video Pipeline can display graphics data stored in the Frame Buffer. This graphic data can come directly from the Video Input Port and display the video, from the CPU or from the Graphic Engine (2D accelerator) to display an image. To use all the functionalities of the Graphic Engine like transparency feature, the image needs to be in RGB656 format, similar to the graphics data.

9. GLOSSARY

Frame Buffer :

The Frame Buffer is an area of memory reserved for graphics and video. This area can be set between 128Kb to 4Mb of memory and the Graphics Engine can work only in this area. Frame Buffer memory is equivalent to a video memory in a none UMA architecture.

Host memory :

Host memory correspond to the system memory, visible by the operationg system.

Data Port :

The Data Port FIFO is used to send data to the Graphics Engine when this data is not directly accessible. The Data Port FIFO is mapped in memory and is accessible using a physical memory address in the Graphics Engine area.

Pixel depth :

The pixel depth is the number of Bytes for one pixel of a graphics area. The pixel depth defines the number of colors managed in this area.

Unified Memory architecture :

Unified Memory Architecture (UMA) is a memory architecture where system memory, graphics memory and video memory is physically grouped inside same memory chip, using the same memory bus.

CRTC :

The Cathodic Ray Tube Controller get data from the Frame Buffer to display them on a monitor or a TV.

Video Pipeline :

Video overlay used to display video on screen.

Video Input Port :

Video input used to capture a video stream inside the Frame Buffer.

Bitblit :

Operation of image (rectangle area) copied inside memory.